

More Metadata Management in AdminUI - GSoC 2017 Project

- [Overview](#)
- [Project Goals](#)
- [Implementation](#)
 - [Manage Modules](#)
 - [Features of the Manage Modules](#)
 - [System Information](#)
 - [New Features](#)
 - [Manage Scheduler](#)
 - [Features of the Manage Modules](#)
- [Resources](#)

Overview

- Primary Mentor: [Unknown User \(dkayiwa\)](#)
- Backup Mentor: [Unknown User \(wyclif\)](#)
- Student: [Unknown User \(suthagar23\)](#)

OpenMRS Legacy Module contains a lot of administrative functionalities which are needed to manage the reference application. Most of this administrative functionalities contain a legacy model and less experience to the users. So OpenMRS Community wanted to migrate those administrative features to the Modern view. More Metadata Management in AdminUI project is one of those projects which are designed to migrate some of these administrative functionalities to the modern view. In the More Meta data Management in AdminUI project will focus on this following functionalities while migrating,

1. Manage modules
2. System Information
3. Manage Scheduler

Those features are implemented as Open Web Apps with the modern view to the users.

Project Goals

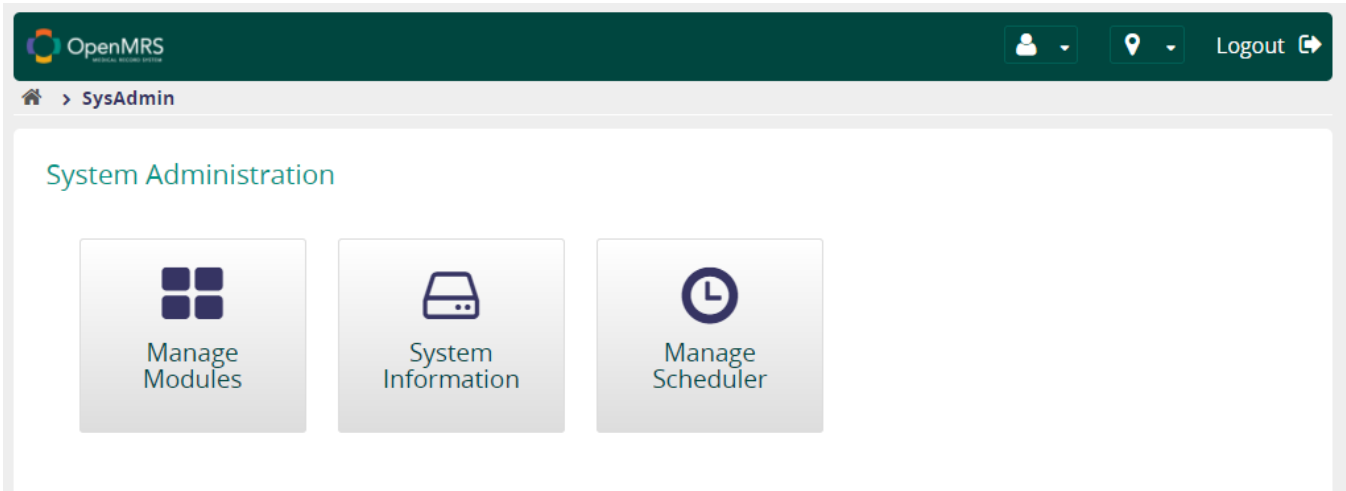
1. Migrate the legacy functionalities to the modern open web apps
 - a. Manage Modules
 - b. System Information
 - c. Manage Schedule
2. Increase the user experience and feasibility of the legacy functionalities
 - a. Use some better user interfaces instead of legacy model
 - b. Change the workflow of the functionalities to increase the usability
 - c. Simplify the administrative functionalities for the users.
3. Extend the usage of the legacy functionalities with REST APIs
4. Extend the administrative features using existing functionalities
5. Improve the problems which are identified in the legacy UI modal
 - a. Implement Module Drag and Drop features
 - b. Implement Module Information page
 - c. Allow user to directly download the modules from OpenMRS Addons
 - d. Increase the System Information view
 - e. Implement Real Time Task Execute features

Implementation

Those features are implemented as an Open Web App and included into the SysAdmin Open Wep App.

Used technologies for the developments,

- Front End Development : HTML, CSS, Angular JS, jQuery
- Back End Development: Java, REST API



Manage Modules

This feature will be used to manage the modules in the OpenMRS reference application. Users can use this implementation for the following functionalities,

Features of the Manage Modules

Functionalities	Modification	New/ Existing/ Modified Feature
1. List all the installed Module	New Icons used to indicate the module status	<i>Modified Feature</i>
2. Start the module	<i>No Modifications</i>	Existing Feature
3. Stop the module	Confirmation Pop up will be shown with the dependent modules details to alert the user	<i>Modified Feature</i>
4. Delete/Unload the module	Confirmation Pop up will be shown with the dependent modules details to alert the user	<i>Modified Feature</i>
5. Check updates	Module updates will be checked with OpenMRS AddOns and listed in the new page for the user selection.	<i>Modified Feature</i>
6. Check one module update	Check the update with OpenMRS AddOns and indicate the update status	New Feature
7. Start All Modules	<i>No Modifications</i>	Existing Feature
8. Add/Upgrade Modules	Implemented Drag and Drop feature	<i>Modified Feature</i>
9. Search Modules	Connected with OpenMRS Addons and user can search the module independently	<i>Modified Feature</i>
10. Search Module Information	User can view the detailed information about the searched module	New Feature
11. Module Information View	Used to display the module information with required modules, aware of modules, and depend on module details	New Feature
12. View not installed module information	Connected with OpenMRS add-ons and indicate the user about the installation features.	New Feature

System Information

This feature will be used to display the System information about the OpenMRS server and the system. Users can use this feature to get the following information,

1. OpenMRS Information
2. Operating System Information
3. Java Runtime Information
4. User Information
5. Memory Information

6. Database Information
7. Module Information

New Features

1. Divided the existing System Information under different set of categories to increase the usability
2. Used some new kind of Icons to illustrate the Information Category properly
3. Modified Module Information Section with some new ideas.

Manage Scheduler

This feature will be used to manage the tasks in the OpenMRS reference application. Users can use this implementation for this following functionalities,

Features of the Manage Modules

Functionalities	Modification	New/Existing/Modified Feature
1. List all the installed Module	New Icons and UI used to indicate the module status	<i>Modified Feature</i>
2. Schedule Task	<i>No Modifications</i>	Existing Feature
3. Shutdown Task	Confirmation Pop up will be shown to alert the user	<i>Modified Feature</i>
4. Reschedule Task	It will reschedule the existing task in the system	New Feature
5. Delete Task	Confirmation Pop up will be shown to alert the user	<i>Modified Feature</i>
6. Reschedule All Tasks	It will reschedule all the tasks in the system	New Feature
7. Shutdown All Tasks	It will shut down all the tasks in the system	New Feature
8. Startup Tasks	It will reschedule all the tasks in the system	New Feature
9. Refresh Tasks	It will refresh the list of registered tasks	Existing Feature
10. Add New Task Definition	Used to create new Task Definition Implemented new UI for this functionality	<i>Modified Feature</i>
11. Edit Task Definition	Used to edit existing Task Definition Implemented new UI for this functionality	<i>Modified Feature</i>

Resources

OpenMRS talk discussions links, (You can get more information about the projects development workflow using those talk threads)

1. [GSoC 2017 - More Metadata Management in AdminUI](#)
2. [REST Implementation for Scheduler in Legacy UI Module](#)
3. [Manage Scheduler Implementation on SysAdmin OWA](#)
4. [New REST Implementation for SystemInformation](#)
5. [System Information Page Interface Modification](#)

Created and worked tickets during the project development

1. OWA-21 : [Implement Module Management Functionalities in OWA](#)
2. OWA-22 : [Implement System Information functionality in OWA](#)
3. OWA-24 : [OWA-22 / Modify System Info Page](#)
4. OWA-25 : [OWA-21 / Implement Module Details view page](#)
5. OWA-26 : [OWA-21 / Integrate with OpenMRS Add-Ons to get module updates](#)
6. OWA-27 : [Implement Manage Scheduler functionalities in OWA](#)
7. RESTWS-673 : [Implement REST services to Manage Scheduler functionalities.](#)
8. RESTWS-674 : [Include More Test cases for ModuleController1_8 in REST Module](#)
9. OWA-31 : [OWA-21 Implement Drag and Drop module upload feature](#)

Created tickets for the issues by me during the development period

1. AO-8 : [Allow cross-origin calls to our REST API](#)
2. AO-9 : [REST endpoint for getting a module by it's modulePackage value](#)
3. PDMO-1 : [Implement Encounters functionalities in OWA](#)

Weekly Blog posts during the development period

Week	Blog Post
Week 1	OpenMRS Module Management OWA
Week 2	Upload module to OpenMRS using REST
Week 3	System Information Open Web App
Week 4	OpenMRS System Administration OWA
Week 5	First Evaluation — Google Summer of Code
Week 6	System Administration OWA — Final Stage
Week 7	Complete the SysAdmin — Final Stage
Week 8	Module Update with OpenMRS Add Ons
Week 9	Second Evaluations — GSoC 2017
Week 10	Started working on Manage Scheduler
Week 11	Drag and Drop Module Implementation
Week 12	

Github Code base: <https://github.com/suthagar23/openmrs-owa-sysadmin>

suthagar23/openmrs-owa-admin

openmrs-owa-admin - Open Web App for OpenMRS Module Management

github.com

